**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

Курсовая работа

По дисциплине «Алгоритмы и структуры данных»

Тема «Компьютерная логическая игра «Атари Го»

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

Пояснительная записка

Р.02069337.21/822-12 ПЗ-01

Листов 11

**Руководитель разработки**:

профессор каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« » 2022 г.

**Исполнитель**:

студент гр. ИСТбд-22

*Грачева Дарья Александровна*

« » 2022 г.

**2022**

**Введение**

Приложение “Атари Го”. Атари Го – игра шашечного типа. Данная игра считается урезанным вариантом известной игры Го. Однако, если Го – игра глубоко стратегическая, то Атари Го, скорее тактическая, комбинационная. Особенность этой игры – фишки ставятся на пересечении линий, ход осуществляется по горизонтали и вертикали. Приложение включает в себя регистрацию и авторизацию, доступ в личный кабинет и саму игру.

В работе выбрана такая структура данных, как массив.

Структура данных — массив была выбрана потому, что эта структура имеет множество преимуществ:

1. Массивы обеспечивают произвольный доступ к элементам. Это ускоряет доступ к элементам по положению;
2. Массивы хранят несколько данных похожих типов с одним и тем же именем;
3. В массиве данные организованны таким образом, что ими легко и удобно манипулировать.

Массив используется для формирования доски.

**1. Проектная часть**

**1.1 Постановка задачи на разработку приложения**

Определяется общей постановкой задачи в задании на курсовую работу. Детализируется в разработанном техническом задании (приложение 1).

**1.2 Математические методы**

Математический аппарат не используется.

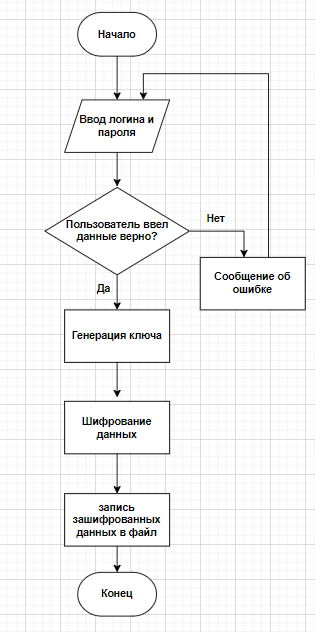
**1.3 Архитектура и алгоритмы**

1.3.1. Архитектура

В приложении были использованы такая структура данных как массив. Функции взаимодействуют между собой, например, в процессе выполнения функции хода вызываются функции проверки на победу.

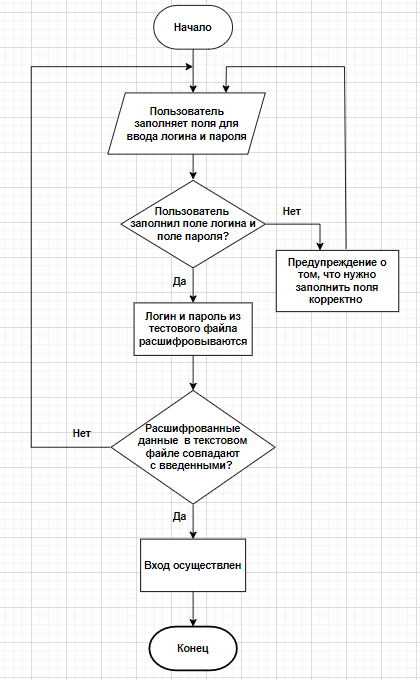
1.3.2 Алгоритм Шифрование

Алгоритм выполняет регистрацию пользователя и шифрование введенных данных пользователя в текстовый файл .txt. Пользователь вводит логин и пароль в форму, после чего идет проверка на пустые строки в случае, если пользователь ничего не ввел, появится строка с предупреждением, если же поля логина и пароля заполнены пользователем, то данные шифруются и записываются в текстовый файл.



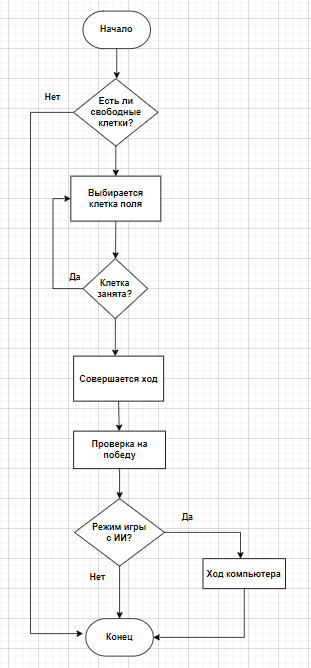
1.3.3 Алгоритм Дешифрование

Данный алгоритм осуществляет авторизацию пользователя и дешифрацию данных из текствого файла. Пользователь вводит логин и пароль в форму, если он ничего не ввел, появляется строка с прдупреждением о том, что необходимо заполнить поля корректно. Если пользователь заполнил поле логина и поле пароля, то наичнается проверка на корректность введенных данных, происходит расшифровка. Данные сверяются, если расшифрованные данные в текстовом файле совпадают с введенными, то осуществляется вход в личный кабинет.



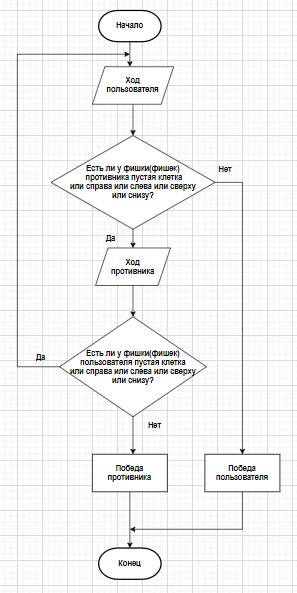
1.3.4 Алгоритм Проверка доступности хода

Данный алгоритм выполняет проверку на доступность хода для камня (фишки). Алгоритм проверяет, есть ли свободные места для размещения фишки. Далее выбирается клетка поля, если клетка не занята, то совершается ход. Далее осуществляется функция проверки на победу. Если режим игры с использованием компьютера (ИИ), то далее совершается ход компьютера.



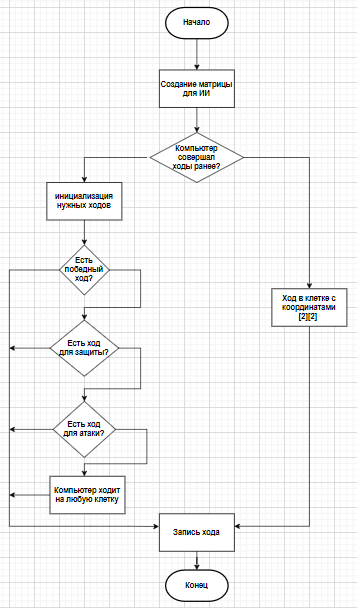
1.3.5 Алгоритм Процесс хода и победа

Данный алгоритм обрабатывает процесс хода игрока. Алгоритм проверяет остается ли у противника дамэ (хотя бы одна клетка сверху или снизу или справа или слева от заблокированной фишки). Если дамэ есть, то ход противника. Если нет, то пользователь выигрывает.



1.3.6 Алгоритм Оценочная функция

Алгоритм оценочной функции позволяет компьютеру ходить в соответствии с правилами игры и стараться выиграть у пользователя.



1.3.7 Алгоритм регистрации

Данный алгоритм осуществляет регистрацию пользователя, зашифровывая данные с помощью алгоритма шифрования и записывая данные в текстовый документ .txt

1.3.8 Алгоритм авторизации

Алгоритм осуществляет авторизацию пользователя. Пользователь вводит логин и пароль, а затем алгоритм проверяет зарегистрирован ли такой пользователь.

1.3.9 Алгоритм выхода из игры

Алгоритм осуществляет выход пользователя из игры по нажатию на крестик в верхнем правом углу.

**1.4 Тестирование**

Для проверки реализованного приложения проведено ручное тестирование. Исходный код проверен на наличие неиспользуемых переменных, ошибок, связанных с логикой игры, ошибок форматирования введенных данных пользователем.

Для тестирования взяты два распространённых сценария наиболее часто приводящих к ошибке.

1 Сценарий. Неверный ввод пользователем данных.

" Действие пользователя: пустые поля логина и пароля"

"Ожидаемый результат: пользователь получит сообщение "\*используйте от 4 до 14 символов""

"Фактический результат: пользователь получает сообщение "\* используйте от 4 до 14 символов ""

2 Сценарий. Попытка авторизации пользователя, ранее не зарегистрированного.

" Действие пользователя: авторизация несуществующего аккаунта"

"Ожидаемый результат: пользователь получит сообщение ""\*неверный логин или пароль""

"Фактический результат: пользователь получает сообщение ""\*неверный логин или пароль"".

**2. Источники, использованные при разработке**

1. Шашки – Атари Го [Электронный ресурс]: Шашки – Атари Го – URL: http://lotos-khv.ru/game/games/atari.pdf (дата обращения: 09.01.2023)
2. Школа Го [Электронный ресурс]: Школа Го – URL: http://go-igo.ru/dictionary-go/atari.html (дата обращения: 09.01.2023)

**3. Приложения**

1. Приложение 1. Техническое задание.
2. Приложение 2. Руководство программиста.